УДК 004.4

Сравнение стратегий ветвления в системе контроля версий

Мигулаева Т. А. (Университет ИТМО)
Научный руководитель – аспирант, Филянин И. В.
(Университет ИТМО)

Введение. Системы контроля версий дают разработчикам программного обеспечения возможность систематизировать свой код, отслеживать его состояние и управлять им.
В частности, системы контроля версий помогают разработчикам работать над кодом проекта в команде. Сочетание функций, таких как коммиты и ветки, с конкретными принципами и стратегиями помогает командам организовать код и сократить время, необходимое для управления версиями кода.[1]
От потребностей разработчика, команды и проекта зависит то, каким именно способом будет организован подход к разработке и внедрению кода. Именно поэтому существует несколько стратегий ветвления, каждая из которых имеет свои положительные и отрицательные. Изучение этих стратегий актуально в связи с возрастающими потребностями в скорости и сложности разработки.

Основная часть. В ходе развития систем контроля версий и популяризации использования этих систем в разработке появилось несколько стратегий ветвления, решающих различные задачи разработки. Используя различные из этих систем, разработчики получают возможность ускорять разворачивание кода, увеличивать качество проверки каждого внесённого изменения или увеличивать количество членов команды до нескольких тысяч человек. [2]
На данный момент нерешённым вопросом является вопрос выбора наилучшей стратегии ветвления, поскольку исходя из требований каждого конкретного проекта и возможностей команды разработчиков результат может различаться.
Для исследования были выбраны самые популярные стратегии ветвления в системе контроля версий. [3] В описание каждой системы ветвления входит модель её использования, что даёт возможность предсказать удобство выбора конкретного способа ветвления для отдельного проекта. Для исследования были выбраны самые популярные стратегии ветвления в системе контроля версий. В рамках исследований были проведены несколько экспериментов, обосновывающих использование данных инструментов в конкретных проектах.

Выводы. В результате исследования и проведения эксперимента был проведён сравнительный анализ выбранных систем, в результате которого были выявлены достоинства и недостатки каждой системы. По результатам анализа сформированы критерии сопоставления свойств каждой из систем ветвления с характеристиками различных типов проектов для получения требуемого результата.

Список использованных источников:

1. Scott Chacon, Ben Straub. Pro Git. – 2009. – С. 63–85.
2. Paul Hammant, Steve Smith Trunk Based Development. – 2017. – С. 121–148.
3. Emma Jane Hogbin Westby, Git for Teams: A User-Centered Approach to Creating Efficient Workflows in Git. – 2015. – С. 33–51.